

BAB III

DASAR TEORI

3.1 Sistem Pendukung Keputusan (SPK)

Pada saat memasuki abad 21, terjadi perubahan besar bagaimana para manajer menggunakan dukungan komputerisasi dalam pengambilan keputusan dengan menggunakan sistem pendukung keputusan. SPK digunakan sebagai alat bantu bagi para pengambil keputusan untuk memperluas kapabilitas para pengambil keputusan, namun tidak untuk menggantikan penilaian para pengambil keputusan. SPK ditujukan untuk keputusan-keputusan yang memerlukan penilaian atau untuk keputusan-keputusan yang sama sekali tidak dapat didukung oleh algoritma. SPK meluas dengan cepat, dari sekadar alat pendukung personal menjadi komoditas yang dipakai bersama (Turban, Aronson dan Liang, 2006). Karakteristik dan kemampuan dari SPK antara lain (Turban, 2005):

1. Membantu manajer dalam pengambilan keputusan atas masalah semi terstruktur.
2. Memberikan dukungan atas pertimbangan manajer dalam level yang berbeda-beda.
3. Meningkatkan efektivitas keputusan yang diambil manajer lebih daripada perbaikan efisiensinya.
4. SPK mudah digunakan karena mempunyai fleksibilitas dan tampilan grafik yang bagus serta bersifat interaktif.
5. Dukungan kualitas. Komputer bisa meningkatkan kualitas keputusan yang dibuat.

6. SPK adaptif sepanjang waktu. Karena fleksibel sehingga dapat ditambah, diubah, dikombinasikan, dan dapat diatur kembali elemen dasarnya.

Disamping berbagai tujuan diatas, Sistem Pendukung Keputusan (SPK) juga memiliki keterbatasan, antara lain (Turban, 2005):

- 1) Ada beberapa kemampuan manajemen dan bakat manusia yang tidak dapat dimodelkan, sehingga model yang ada dalam sistem tidak semuanya mencerminkan persoalan sebenarnya.
- 2) Kemampuan suatu SPK terbatas pada pembendaharaan pengetahuan yang dimilikinya (pengetahuan dasar serta model dasar).
- 3) Proses-proses yang dapat dilakukan SPK biasanya tergantung juga pada kemampuan perangkat lunak yang digunakannya.
- 4) Sistem Pendukung Keputusan tidak memiliki kemampuan intuisi seperti yang dimiliki manusia. Karena walaupun dan bagaimana pun canggihnya suatu SPK, dia hanyalah suatu kumpulan perangkat keras, perangkat lunak, dan sistem operasi yang tidak dilengkapi dengan kemampuan berpikir.

Secara luas, SPK dirancang untuk menghasilkan berbagai alternatif yang ditawarkan kepada para pengambil keputusan dalam melaksanakan tugasnya karena sebagian besar proses pengambilan keputusan yaitu perumusan masalah, dan pencarian alternatif yang telah dikerjakan oleh sistem maka diharapkan pengambilan keputusan akan lebih cepat dan akurat dalam menangani masalah yang dihadapi.

3.2 Proses Pengambilan Keputusan

Menurut Simon (1977), terdapat tiga fase dalam proses pengambilan keputusan (Turban, Aronson dan Liang, 2006):

1. Fase Inteligensi (*Intelligence*)

Dalam tahap intelligence tersebut seseorang dalam rangka pengambilan keputusan untuk permasalahan yang dihadapi yang terdiri dari aktivitas penelusuran, pendeteksian serta proses pengenalan masalah. Data yang diperoleh diproses dan diuji dalam rangka mengidentifikasi masalah.

2. Fase Desain (*Design*)

Tahap proses pengambilan keputusan setelah tahap intelegensi meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi. Aktivitas yang biasanya dilakukan seperti menemukan, mengembangkan dan menganalisa alternatif tindakan yang dapat dilakukan.

3. Fase Pilihan (*Choice*)

Pada tahap ini seorang pengambil keputusan melakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan.

3.3 Pembangunan Sistem Pendukung Keputusan (SPK)

Ada beberapa *tools* yang dapat digunakan dalam pembangunan Sistem Pendukung Keputusan (SPK), yaitu (Srinivasan, 2000):

1) *Database relasional*

Database relasional digunakan jika dataset mendasari komponen penting dalam sistem pembuat keputusan.

2) *Spreadsheets*

Spreadsheets digunakan jika data ditampilkan dalam bentuk format kolom dan baris.

3) *Bahasa pemrograman*

Bahasa pemrograman digunakan jika model menjadi kompleks maka dibutuhkan bahasa pemrograman yang sesuai. Strategi pengembangannya adalah menggunakan bahasa pemrograman generik atau umum, seperti C, Visual Basic, Delphi, atau Pascal.

4) *Sistem objek relasional*

Sistem ini mengkombinasikan kemampuan penanganan data dari database relasional dengan spesifikasi detail dari bahasa pemrograman.

5) *Online Analytical Processing System (OLAP)*

Online Analytical Processing System (OLAP) adalah tools yang memudahkan untuk spesifikasi dan penerimaan data dimana terdapat banyak kriteria data.

6) *Data warehouse*

Konsep ini digunakan jika banyak data yang harus diorganisasikan untuk kegunaan dimasa yang akan datang.

7) *Interface dan integration tools*

Beberapa tools yang digunakan untuk mempermudah mekanisme pengguna berinteraksi dengan sistem.

3.4 Arsitektur Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan (SPK) terdiri dari beberapa subsistem, yaitu (Turban, 2005):

1. Subsistem Manajemen Data

Subsistem manajemen data memasukkan satu *database* yang berisi data yang relevan untuk situasi dan dikelola oleh perangkat lunak yang disebut *Data Base Management System (DBMS)*.

2. Subsistem Manajemen Model

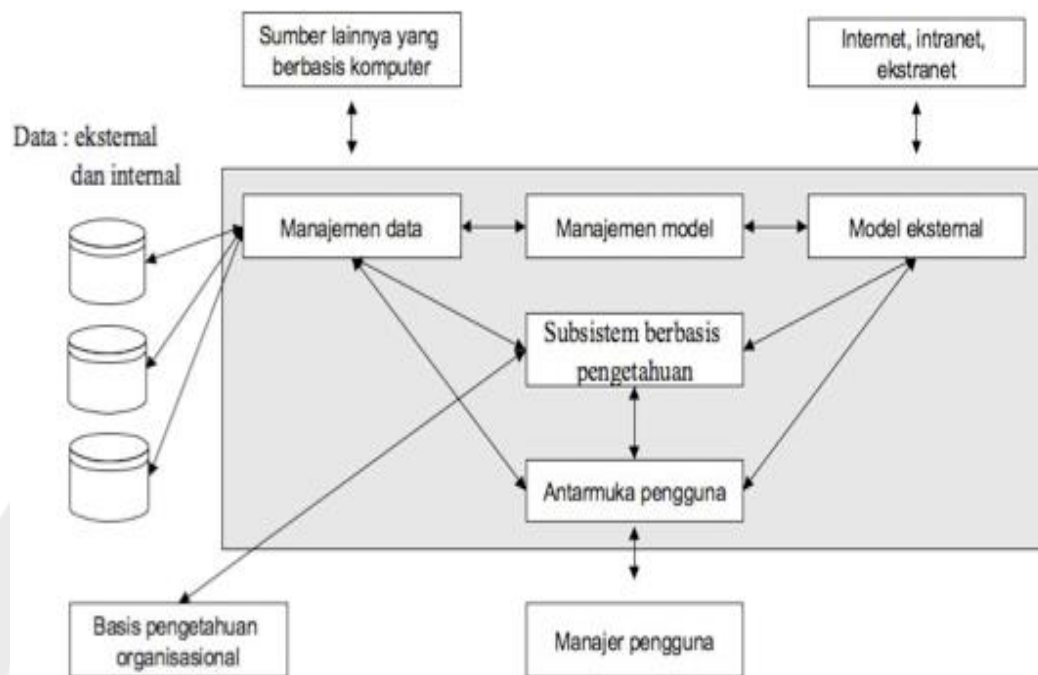
Subsistem manajemen model merupakan paket perangkat lunak yang memasukkan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lain yang memberi kapabilitas analitik dan manajemen perangkat lunak yang tepat.

3. Subsistem Manajemen Pengguna

Subsistem manajemen pengguna yaitu pengguna yang berkomunikasi dan memerintahkan Sistem Pendukung Keputusan (SPK) melalui subsistem tersebut. Pengguna adalah bagian yang dipertimbangkan dari sistem.

4. Subsistem Manajemen Berbasis Pengetahuan

Subsistem tersebut mendukung semua subsistem lain atau bertindak langsung sebagai suatu komponen independent dan bersifat opsional.



Gambar 3. 1 Sistem Pendukung Keputusan (SPK) (Turban, 2005)

3.5 K-Nearest Neighbour

K-Nearest Neighbour merupakan salah satu metode yang digunakan dalam pengklasifikasian. Prinsip kerja *K-Nearest Neighbour (KNN)* adalah mencari jarak terdekat antara data yang akan dievaluasi dengan *K*-tetangga (*neighbour*) terdekatnya dalam data pelatihan (Kusumadewi, 2008). Algoritma *K-Nearest Neighbour (KNN)* adalah sebuah metode yang melakukan klasifikasi terhadap obyek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Prinsip kerja dari *K-Nearest Neighbour (KNN)* adalah mencari jarak terdekat antara data yang akan dievaluasi dengan *K* tetangga (*neighbour*) terdekatnya dalam data pelatihan. Pada fase pembelajaran, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data pembelajaran. Pada klasifikasi vektor yang sama dihitung untuk data test (yang klasifikasinya tidak diketahui).

Jarak dari vektor yang baru ini terhadap seluruh vektor data pembelajaran dihitung, dan sejumlah K buah yang paling dekat diambil. Titik yang baru klasifikasinya diprediksi termasuk pada klasifikasi terbanyak dari titik-titik tersebut. Nilai K yang terbaik untuk algoritma ini tergantung pada data. Secara umum, nilai k yang tinggi akan mengurangi efek *noise* pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi lebih kabur. Nilai k yang bagus dapat dipilih dengan optimasi parameter, misalnya dengan menggunakan *cross-validation*. Kasus khusus dimana klasifikasi diprediksi berdasarkan data pembelajaran yang paling dekat (dengan kata lain, k=1) disebut algoritma *K-Nearest Neighbour*.

Ketepatan algoritma *K-Nearest Neighbour* ini sangat dipengaruhi oleh ada tidaknya fitur-fitur yang tidak relevan, atau jika bobot fitur tersebut tidak setara dengan relevansinya terhadap klasifikasi. Riset terhadap algoritma ini sebagian besar membahas bagaimana memilih dan member bobot terhadap fitur agar performa klasifikasi menjadi lebih baik. Sesuai dengan prinsip kerja *K-Nearest Neighbour* yaitu mencari jarak terdekat antara data yang akan dievaluasi dengan k-tetangga terdekatnya dalam data pelatihan. Persamaan dibawah ini menunjukkan rumus perhitungan untuk mencari jarak terdekat dengan d adalah jarak dan p adalah dimensi data (Agusta, 2007):

$$d_i = \sqrt{\sum_i^p (x_{2i} - x_{1i})^2}$$

Persamaan *K-Nearest Neighbour*

Dengan

keterangan :

X1 = sampel data

d = jarak

X2 = data uji

p = dimensi data

i = variable data

3.6 Fitness Center

Fitness center, atau dalam Bahasa Indonesianya, 'Pusat Kebugaran' biasa lebih dikenal dengan istilah 'Gym' Adalah suatu tempat untuk menampung berbagai macam alat-alat latihan kesehatan untuk mengadakan kegiatan latihan fisik. *Fitness Center* merupakan suatu layanan di bidang kesehatan yang membantu masyarakat umum meningkatkan kesehatan fisik. Dimana tidak hanya masyarakat menjadi lebih sehat, tapi juga lebih termotivasi, memiliki kepercayaan diri dan semangat hidup.

3.6.1 Fasilitas

Biasanya sebuah *fitness center* memiliki ruang latihan utama, dimana terdapat alat-alat berat untuk berlatih, seperti *dumbell*, *benchpress*, *barbell*, dan semacamnya. Biasanya juga disediakan cermin di ruang ini dimana supaya orang yang berlatih dapat memperhatikan gerak tubuhnya dan posturnya dalam berlatih sudah benar atau belum.

Sebuah *fitness center* biasanya juga menyediakan sebuah ruang latihan *Cardio*, yaitu ruang yang berisi umumnya alat-alat latihan untuk melatih gerak dan jantung, seperti *Threadmill*, *Row machine*, dan *Exercise Bike*. Sekarang ini sebuah *fitness center* juga memiliki

fasilitas ruang untuk melakukan kegiatan bersama seperti ruang *Aerobic*, untuk Yoga, dan kegiatan semacamnya.

Apabila sebuah *fitness center* memiliki modal yang cukup maka dapat juga menyediakan fasilitas untuk Olahraga, seperti lapangan tenis, kolam renang, *boxing*. Biasanya untuk menggunakan fasilitas ini tentu saja pelanggan akan dikenai biaya tambahan. Fasilitas tambahan lainnya bisa berupa restoran, *café*, spa, sauna dan mandi uap.

3.7 Internet

Interconnect Network atau yang lebih populer dengan sebutan *Internet* adalah sebuah sistem komunikasi global yang menghubungkan komputer-komputer dan jaringan-jaringan komputer di seluruh dunia. Setiap komputer dan jaringan terhubung secara langsung maupun tidak ke beberapa jalur utama yang disebut *internet backbone* dan dibedakan satu dengan yang lain menggunakan *unique name* yang biasa disebut dengan alamat IP 32 bit.

3.8 Pengertian Framework

Framework adalah sekumpulan fungsi, class, dan aturan-aturan. Berbeda dengan *library* yang sifatnya untuk tujuan tertentu saja, *framework* bersifat menyeluruh mengatur bagaimana membangun sebuah aplikasi. *Framework* memungkinkan membangun aplikasi dengan lebih cepat. Karena *framework* telah menyediakan hal-hal penunjang lainnya seperti koneksi *database*, *form validation*, GUI, dan *security*.

3.9 Codeigniter

CodeIgniter adalah salah satu framework PHP yang pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc, sebuah perusahaan yang memproduksi CMS *Expression Engine* (Basuki, 2010). *CodeIgniter* menerapkan kaidah MVC (*Model-View-Controller*). MVC digunakan untuk memisahkan komponen-komponen presentasi suatu aplikasi dengan komponen logika bisnisnya (A. Nugroho, 2009). Pada model MVC, Model merupakan tempat kode program yang digunakan untuk memanipulasi *database*, View yang berupa template *HTML/XHTML* atau *PHP* untuk menampilkan data pada *browser* sedangkan *Controller* adalah tempat kode program untuk mengontrol aliran aplikasi (sebagai pengontrol *Model* dan *View*). Terdapat 3 jenis komponen yang membangun suatu *MVC pattern* dalam suatu aplikasi yaitu:

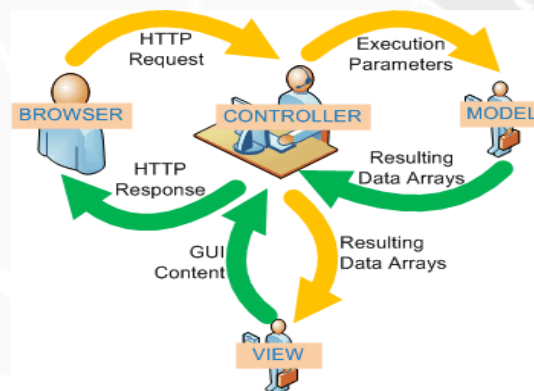
View, merupakan bagian yang menangani *presentation logic*. Pada suatu aplikasi web bagian ini biasanya berupa file template *HTML*, yang diatur oleh *controller*. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.

Model, biasanya berhubungan langsung dengan *database* untuk memanipulasi data (*insert, update, delete, search*), menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan bagian *view*.

Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, *controller* berfungsi untuk menerima request dan data dari user

kemudian menentukan apa yang akan diproses oleh aplikasi.

Sehingga skrip *PHP*, *query MySQL*, *Javascript* dan *CSS* bisa dibuat terpisah tidak dalam satu skrip berukuran besar untuk memudahkan proses *maintenance* dan pengembangan lebih lanjut (Basuki, 2010). Misalnya bila ingin mengubah tampilan aplikasi, maka kita cukup mengubah kode program yang berada pada View saja tidak perlu mengubah kode program secara keseluruhan.



Gambar 3. 2 Arsitektur MVC pada Codeigniter (Betha Sidik, 2012)

3.10 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) adalah salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. Sama halnya *styles* dalam aplikasi pengolahan kata seperti *microsoft word* yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images* dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). pada umumnya *Cascading Style Sheets (CSS)* dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa *HTML* dan *XHTML* (suryatiningsih, 2009).

3.11 Konsep Basis data

Model basis data menunjukkan suatu cara yang ditujukan untuk mengelola jaringan data secara fisik dalam memori sekunder yang berdampak pada bagaimana mengelompokkan dan membentuk keseluruhan data yang terkait kedalam sistem yang sedang ditinjau.

3.11.1 Abstraksi Data

Abstraksi data merupakan tingkatan dalam melihat data sebuah system basis data. Abstraksi data terbagi dalam 3 level yaitu:

- a. Level Fisik (*physical level*) merupakan level terendah dalam abstraksi data yang menunjukkan bagaimana suatu data disimpan.
- b. Level logika/konseptual (*conceptual level*) merupakan level yang lebih tinggi dari pada level fisik, yang menggambarkan data apa sebenarnya (secara fungsional) disimpan dalam basis data dan hubungannya dengan data lain.
- c. Level penampakan (*view level*) merupakan level tertinggi dari abstraksi data, yang hanya menunjukkan sebagian dari basis data.

3.11.2 Key

Key adalah satu gabungan beberapa atribut yang dapat membedakan sebuah entitas dengan entitas lain. Beberapa macam key antara lain adalah:

- a. *Superkey* merupakan satu atau lebih atribut (kumpulan atribut) yang dapat membedakan sebuah entitas didalam sebuah himpunan entitas.

- b. *Candidate key* adalah superkey yang tidak mengandung superkey lainnya, yang merupakan subset dari superkey pertama.
- c. *Primary key* adalah atribut yang dapat digunakan untuk membedakan sebuah entitas dalam sebuah himpunan entitas.

3.11.3 Hypertext Preprocessor (PHP)

Hypertext Preprocessor adalah *server side scripting enviroment* yang dapat digunakan untuk membuat dan menjalankan aplikasi-aplikasi di *web server* agar lebih interaktif dan *programmable*. Dengan adanya PHP, aplikasi-aplikasi yang ada di *web server* benar-benar dijalankan di *web server* tanpa mengharuskan adanya tambahan atau syarat tertentu untuk sisi klien (*web browser*). Sebagian besar sintaks mirip dengan bahasa C, Java, Asp dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama bahasa ini adalah untuk memungkinkan perancang web untuk menulis halaman web dinamik dengan cepat. Kelebihan dari bahasa pemrograman ini adalah seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan yang paling utama PHP adalah pada konektivitasnya dengan sistem database di dalam web. (Kadir, 2009).

3.12 MySQL

MySQL merupakan aplikasi *SQL Database Server* yang *multi-user* dan *multithread* serta *open source* yang telah mempunyai lisensi *GNU Public Lisence (GPL)*. *MySQL* merupakan salah satu *software database* (basis data) *open source* dan juga *enterprise* yang dikembangkan sebuah komunitas bernama *MySQL AB* dengan tujuan membantu user

untuk menyimpan data dalam tabel-tabel. Tabel-tabel tersebut dibungkus dalam suatu wadah yang biasa disebut dengan *database*. *MySQL* dapat menampung banyak *schemata*, dimana masing-masing *schemata* ini dapat digunakan oleh aplikasi yang berbeda-beda. *MySQL* merupakan *database* yang bersifat online sehingga dapat digunakan untuk aplikasi *Multi User* (Maghfirah, 2010).

3.13 XAMPP

XAMPP adalah distribusi *apache* yang kecil dan ringan yang mengandung teknologi pengembangan web yang paling umum dalam satu paket. Isi, ukuran kecil, dan mudah dibawa membuat alat ini ideal untuk siswa dalam mengembangkan dan menguji aplikasi di *PHP* dan *MySQL*. *XAMPP* tersedia bebas untuk di *download* untuk dua paket tertentu : *full* dan *lite*. Sementara *download* paket *full* menyediakan berbagai alat pembangunan. Sesuai namanya, versi *lite* adalah paket kecil yang berisi *Apache HTTP Server*, *PHP*, *MySQL*, *phpMyAdmin*, *Openssl*, *SQLite*. Semua paket *XAMPP* dan *add-ons* didistribusikan melalui *Apache Friends* di alamat website <http://www.apachefriends.org/> (Dvorski, 2007).

3.14 Data Base Management System (DBMS)

Data Base Management System (DBMS) merupakan sistem yang secara khusus dibuat untuk memudahkan pemakai dalam mengelola basis data. *Data Base Management System (DBMS)* memiliki keuntungan seperti berikut : Independensi data, pengaksesan yang efisien terhadap data, keamanan dan integritas data, administrasi data yang akan mengakibatkan berkurangnya redundansi atau duplikasi data, akses bersamaan dan pemulihan terhadap kegagalan,

waktu pengembangan aplikasi diperpendek (Kadir dan Triwahyuni, 2003).

3.15 Peta

Peta adalah suatu yang digunakan untuk merepresentasikan permukaan dari suatu bidang lengkung pada suatu bidang datar (Puntodewo, dkk, 2003). Dalam sistem layanan berbasis lokasi petanya direpresentasikan menjadi obyek spasial dimana dalam obyek spasial menggunakan 2 mode penggambaran peta yaitu mode raster dan mode vektor. Mode raster adalah gambar peta yang terdiri dari sejumlah pixel yang terorganisir dalam sebuah grid baris dan kolom. Mode vektor adalah gambar peta yang ditentukan oleh koordinat dari suatu sistem yang menjadi referensi (Kupper, 2005).

3.16 Google Maps API

Google Maps Api digunakan untuk mengetahui posisi tidak hanya dengan mengetahui koordinatnya saja, tetapi juga dari posisi tersebut akan ditampilkan dalam sebuah gambaran peta. Salah satunya adalah peta *Google*. Peta *Google* dapat diubah, diperbesar. Peta *Google* mengandung unsur teks, gambar, *link*, dan mengarah pada aplikasi lain (Workman, Gschwender, dan Chan 2005). Dengan memakai "*Google Map API*" akan memberikan kemampuan pada seorang pembuat program untuk bisa menambahkan peta *Google* pada aplikasi mereka (Trahan et.al. 2008).